C. Remarks

In the office action, claims 1-5, 7-16, 20-24, 27-40, and 44-48 were rejected under 35 U.S.C. 102 (e) as being anticipated by the patent to Abrams et al. U.S. patent application number: 2002/0166117 A1 (herein after referred to as, "Abrams"). Further, claims 1 was rejected under 35 U.S.C. 112 (first paragraph) as not being able to enable any person skilled in the art to which it pertains, or with which it is most nearly connected, to make and use the invention commensurate in scope with the claims. Claim 1 was also rejected under 35 U.S.C 112 (second paragraph) as being indefinite for filing to particularly point out and distinctly claim the subject matter. The indefiniteness has been corrected in the claim amendments.

The present invention provides a method and system for handling a request for a resource. The resource is part of a plurality of resources that are encapsulated in symbionts. Further, the symbionts run in host programs. The host programs run on computers connected in a network. Applications running on a computer make a request for a resource. Thereafter, a host program comprising a symbiont encapsulating the resource, receives the request. The symbiont then handles the request by performing one of the following actions depending on certain conditions:

a. serving the request.
b. replicating itself on to the *computer* that has the applications running on it.
c. redirecting the request to an adjoining symbiont.

The applications can make the request through a host program running on the computer. The symbiont then performs the step of replication by replicating itself onto the computer.

In contrast, Abrams teaches a system and method for providing distributed, on-demand application processing. Abrams relates to a method and system that are primarily applicable for client-server architecture. The system includes a set of computer resources, a resource manager and at least one restored module. Further, multiple instances of an application are made available on a server. The system includes a resource manager which handles the multiple instances of an application. However, in the present invention each symbiont and host program is responsible for handling requests for resources. Hence there are multiple units which handle the requests in place of the resource manager.

<u>35 U.S.C. 112 rejections</u>

Regarding the 35 U.S.C. 112 (first paragraph) enablement rejection, at the outset it is noted that all patent applications are directed at persons of ordinary skill in the art to which it pertains. As such, matters that these persons would have knowledge of need not be described in a patent application directed to such common knowledge. In the case at hand a person of ordinary skill in the art would have knowledge of how computer networks operate and how to write control software for the same. In the office action, claim 1 was rejected as not being able to provide enablement for "...and said host has not been redirected more than a *predetermined* number of times". Making computer programs run a certain routine a predetermined (limited) number of times before branching to another action is very well known to persons of ordinary skill in the art of computer

programming and network control. Thus, the predetermined number may be any number that a person skilled in the art would specify. This predetermined number may be modified depending on the performance of the network by any person skilled in the art. Further, this predetermined number only affects the efficiency of the method and not its functionality, as claimed. That is, the method, as claimed, would be functional with any value assigned to this 'predetermined number'. This predetermined number may be used to improve the efficiency of the method by any person skilled in the art. Accordingly the present application and claim 1 are well enabled to any person of ordinary skill in this art and it is respectfully requested that the 35 U.S.C. 112 (first paragraph) enablement rejection be removed..

Regarding the 35 U.S.C. 112 (second paragraph) indefiniteness rejection, in order to more clearly define and distinctly claim the present invention, the independent claims 1 and 12 have been amended to recite that the method and system of the present invention relates to handling a request for resource on a network of computers and that the replication takes place on a *computer* running on the network.. Accordingly it is respectfully requested that the 35 U.S.C. 112 (second paragraph) indefiniteness rejection be removed

<u>Prior art rejections</u>

According to the method of the present invention, as set forth in amended claim 1 and claim 12, the present invention may be implemented on any type of network architecture between computers. However, Abrams's system is practiced primarily on client-server architecture. In the present invention, each computer of a network may

comprise a host program, and a host program may comprise a self replicating program called a symbiont. Therefore, the present invention does not distinguish between a computer that has a host program and a computer that has applications running on it. Symbionts may be replicated onto the host program that makes the request. However, according to Abrams's system multiple instances (replicates) of an application are created on the host that serves the request (edge point). Please refer to page 8, paragraph [0073].

The symbiont of the present invention is a self replicating program that encapsulates a resource, such as a network resource or computational resource. Abrams's instances of applications are replicates of the applications requested by a client. Further, the symbiont can perform any one of the following steps in respect of a resource: serving a request, redirecting a request or replicating the resource on to a requesting host. Therefore, in the present invention each symbiont is an intelligent decision making component, responsible for resource allocation and distribution. The support for these recitations is found at page 10, lines 26-29, page 12, lines 11-25 and page 13, lines 1-9. It may be noted that as per Abrams's system, there is a single resource manager. This resource manager is the decision making component that performs the management of distributed resources.

Further, according to the present invention, the replication of a symbiont can occur on a requesting host. Thus the load is shared by servers as well as the clients onto which the resources are replicated. However, according to Abrams, the replication of an application occurs only on the interconnected servers. Hence only the servers share the load for resources. Moreover, Abrams's patent application does not mention that a

resource can be replicated on a requesting client. It is thus apparent that the present invention offers flexibility, scalability, and robustness. Further, the present invention may be implemented without any dedicated hardware as both the system and the host are embodied in the form of software components. The support for this recitation is found at page 10, lines 13-15.

Dependent claim 2, 13, 21, and 37

As to claims 2, 13, 21, and 37, the present invention teaches that, the computers in the network comprise host programs, which expose the symbionts in the network, to the applications running on a computer. The applications on a computer are therefore aware of the resources that are present on the network. That is, the applications on a computer are aware of the resources available on the network before they make the request for any of the available resources. This enables the host to manage the communication with the symbionts in a more effective manner as the host would know *a priori* about the symbionts that have the requisite resource. The support for this recitation is found on page 11, lines 2-3. However, in the case of Abrams's invention, the host program (edgepoint) receives the request. After receiving the request, the host program determines which symbiont (aapshot) provides the desired resource (page 8, paragraph [0073]). Hence the method according to the present invention would be able to handle the request efficiently as the applications would know *a priori* about the availability of resources.

Dependent claims 3, 14, 22, and 38

As to claims 3, 14, 22, and 38, Abrams teaches that a server exposes resources on the client to the network. In the present invention, a computer in the network that comprises a host program exposes a symbiont stored within the host program to the network. This feature also enables each host program in the network to have information regarding all the symbionts present in the network. Therefore, this feature optimizes the time taken for the host program to find an appropriate symbiont that can serve its request. In Abrams's patent application, each host program in the network is provided about the information regarding a symbiont that is present on the network only when the host program makes a request for the symbiont. Please refer to page 8, paragraph [0073].This results in an additional time lag in handling the request.

Dependent claims 4, 15, 23, and 39

As to claims 4, 15, 23, and 39, Abrams doesn't teach any logical connection between the object replicates. However in the present invention, all replicates of a particular resource are connected together in a logical network. The connection between the resources enables a symbiont in the network to possess the load information about neighboring symbionts. This reduces the time taken for a symbiont to make a decision regarding the method for handling a request. For example, the symbiont will know which of its two neighbors have a less load and therefore best suited for redirection of a request. The support for these recitations is found at page 11, lines 9-24. Abrams's invention teaches a on-demand application processing system in cooperation with the preexisting

internet infrastructure. However, Abram does not teach that the replicates (application instances) are logically connected in a ring. Please refer to page 9, paragraph [0082].

Dependent claims 5, 16, 24, and 40

As to claims 5, 16, 24, and 40, Abrams doesn't describe the type of connection between the replicates of a symbiont. Further, Abrams does not teach the concept of a multiply connected ring. However in the present invention all replicates of a particular resource are connected together in the form of a multiply connected ring. In a multiply connected ring there is a tradeoff between the networking load and the load for resources. This trade off depends on the number of neighbors to which each symbiont is connected. For example, if the number of neighbors to which each symbiont is connected is increased, then the networking load would increase, as the total number of connections between the symbionts would increase. However, the number of redirections that a request suffers decreases, as a symbiont would have the information about the loads on more number of neighbors. Hence the present invention provides the flexibility to alter the tradeoff between the networking load and the load for resources. Abrams's invention does not provide any such flexibility. The support for these recitations is found at page 11, lines 9-24.

Dependent claims 7-9, 28-30

As to claims 7-9 and 28-30, by reducing the threshold lmax, the number of symbionts that can exist on the network can be increased dynamically (according to the load on the symbiont and the network). For instance, if the requirement of a particular

resource is very high, then the value of Imax may be decreased so that a replicate of a symbiont may be created more easily on a requesting host. This results in more number of symbionts. Alternatively the value of Imax may be kept higher to minimize the number of possible replicates of the symbiont. Such an alteration in the value of Imax occurs in real time and not on the basis of the predicted demand of the resource. Also, there would be a time lag between the redirection of a request to a neighboring symbiont and the neighboring symbiont receiving the request. The load on the neighboring symbiont may increase during the time lag. Thus, the load on neighboring symbionts is compared to a different threshold 't'. This threshold t is less than Imax

Abrams does not teach the use of two thresholds, Imax and t or the relation between them. However, the present invention teaches the use of two different thresholds. The value of t is deliberately kept lower than Imax as the load of neighboring symbionts is acquired only after a predefined time intervals by the symbiont. The support for these recitations is found at page 12, lines 11-23 and page 13, lines 1-3.

Dependent claims 10-11

As to claim 10, the present invention recites that a symbiont will redirect the request to the neighboring symbiont with the least load. The support for this recitation is found on page 12, lines 18 – 24. As to claim 11, the symbiont closest to the host making a request for a resource will be used for handling the request. The closeness may be in terms of the geographical distance or any other parameter that may be predefined. The support for this recitation is found on page 15, lines 3-5. Therefore, according to the present invention, the symbiont has the intelligence to choose the neighbor that is best

suited for handling a request. Further, the system is configured such that the symbiont closest to the requesting host handles the request. This step decreases the time taken for a request to be handled, thereby improving the efficiency of the method and system for handling the request. Also, there are multiple copies of the component with intelligence (symbiont) which makes the system robust.

In Abrams system, the host program (edge point) determines which symbiont (appshot) provides the desired resource (application). However, the host program does not have the intelligence to check the load on different symbionts providing the desired application. Further, the host program does not redirect (route) the request to the symbiont with the least load.

Independent claims 20 and 36

As to claim 20 and 36, Abrams teaches a method and system for providing resources (applications) primarily in client-server architecture. The resources (applications) are duplicated on to servers and not the clients. However the present invention defines hosts, symbionts and resources that are not limited to client-server architecture. For example, the invention can be practiced on a peer to peer architecture in a LAN environment. In other embodiments, a WAN or a MAN environment may also be used for the enablement of the invention. The only requirements for the enablement of the invention are the software programs such as the symbionts and the hosts.

The fact that the resources are logically connected in a multiply connected ring makes the network of resources robust. The arrangement also facilitates better sharing of information amongst the symbionts. Further, a symbiont along with the resource is

replicated rather than the resource (application) alone as claimed by Abrams. That is, the intelligence required for handling a request for a resource is replicated making the network of symbionts extremely intelligent software, capable of increasing in scale, as per the demand of resources. Therefore, if the load on a symbiont increases above a threshold value (lmax), the symbiont replicates. Similarly, the symbionts cease to exist if the load on the symbiont falls below a minimum threshold value, thereby keeping the number of symbionts at an optimum level. The support for these recitations is found at page 10, lines 12-29.

Dependent claim 27

In claim 27, the birthing rules are governed by the load on the symbiont handling the request, the number of redirections of a request and the load on the neighboring symbionts. A symbiont is replicated on a requesting host if the load on the symbiont is greater than lmax and the number of redirections exceed a value r_{max}. The value of r_{max} may be any number that a person skilled in the art would specify. This r_{max} may be modified depending on the performance by any person skilled in the art. Further, this r_{max} only affects the efficiency of the method and not its functionality, as claimed. That is, the method, as claimed, would be functional with any value assigned to r_{max}. This value may be used to improve the efficiency of the method by any person skilled in the art. Alternatively, the symbiont is replicated if the load on the symbiont exceeds lmax and the load on the neighboring symbionts exceeds the value t. The support for this recitation is found on page 12, lines 10-15.

In contrast, Abrams links the birthing rules of replicas to resources owned by servers. Further, Abrams doesn't teach the use of two thresholds, lmax and t and the relation between them. However, the present invention teaches the use of two different thresholds. The value of t is deliberately kept lower than lmax as the load of neighboring symbionts is acquired only after a predefined time intervals by the symbiont.

Dependent claims 31 and 44

As to claims 31 and 44, the present invention further elaborates on the process for arranging resources in a network of computers. One of the symbionts is marked immortal so that the intelligence required for handling a request for a resource is never deleted. In the present invention, it is the symbiont that is immortal i.e., the intelligence required for handling a request is kept immortal. Abrams does not teach the marking of any one symbiont as an immortal entity. The support for this recitation is found on page 13, lines 21-24.

Dependent claims 32 and 45

In claims 32 and 45, it is recited that a symbiont may cease to exist if the load on the symbiont falls below a threshold lmin. The load is checked at regular time intervals and it is ensured that the time intervals are not kept too short so as to avoid churning. This ensures that the total number of symbionts is kept at an optimum level. The support for these recitations is found at page 13, lines 16-19. Abrams's method however teaches the termination of an instance of an application when the number of users decreases below a threshold. However, an instance does not check the number of users accessing it

at predetermined time intervals. This may result in frequent termination and replication of resources. It may be noted in light of the aforementioned that the use of an optimal value for the time interval would reduce the frequent termination and replication of resources.

Dependent claims 33–34 and 46-47

In claims 33 and 34, it is recited that the time interval for a symbiont to check its own load depends on the time scale of natural fluctuations in load on a symbiont. Therefore the time intervals depend on the actual load variations on a symbiont measured in real time. The support for these recitations is found at page 13, lines 18-20. Abrams does not teach that the symbionts check the load on themselves at regular time intervals. This would result in frequent termination and replication of symbionts.

Dependent claims 35 and 48

In claims 35 and 48, it is recited that one of the symbionts is marked immortal and never ceases to exist. The symbiont is marked immortal so that the intelligence required for handling a request for a resource is never deleted. The support for this recitation is found on page 13, lines 20-24. Abrams does not teach the marking of any one symbiont as an immortal entity.

The present claims have been amended to highlight the distinctions of the present invention over the prior art and it is respectfully submitted that the claims are now clearly patentable over the art of record, and notice to that effect is earnestly solicited, If the

Examiner has any questions regarding this matter, the Examiner is requested to telephone the applicants' attorney at the numbers listed below prior to issuing an Advisory Action.

Respectfully Submitted,

By

William L. Botjer
Reg. No. 27,990
PO Box 478
Center Moriches, NY 11934
(212) 737-5728 (Tue-Thurs)
(631) 874-4826 (Mon & Fri)
(631) 834-0611 (cell if others busy)

Dated:February 21, 2006